PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 5 : G06F 7/52

A1

(11) International Publication Number: WO 91/20028

(43) International Publication Date: 26 December 1991 (26.12.91)

(21) International Application Number: PCT/SE91/00384

(22) International Filing Date: 31 May 1991 (31.05.91)

(30) Priority data:
9002124-7 15 June 1990 (15.06.90) SE

(71)(72) Applicant and Inventor: MASTROVITO, Edoardo [IT/SE]; Fårullsvägen 73, S-583 21 Linköping (SE).

(74) Agent: H W BARNIESKE PATENTBYRÅ AB; P.O. Box 25, S-151 21 Södertälje (SE).

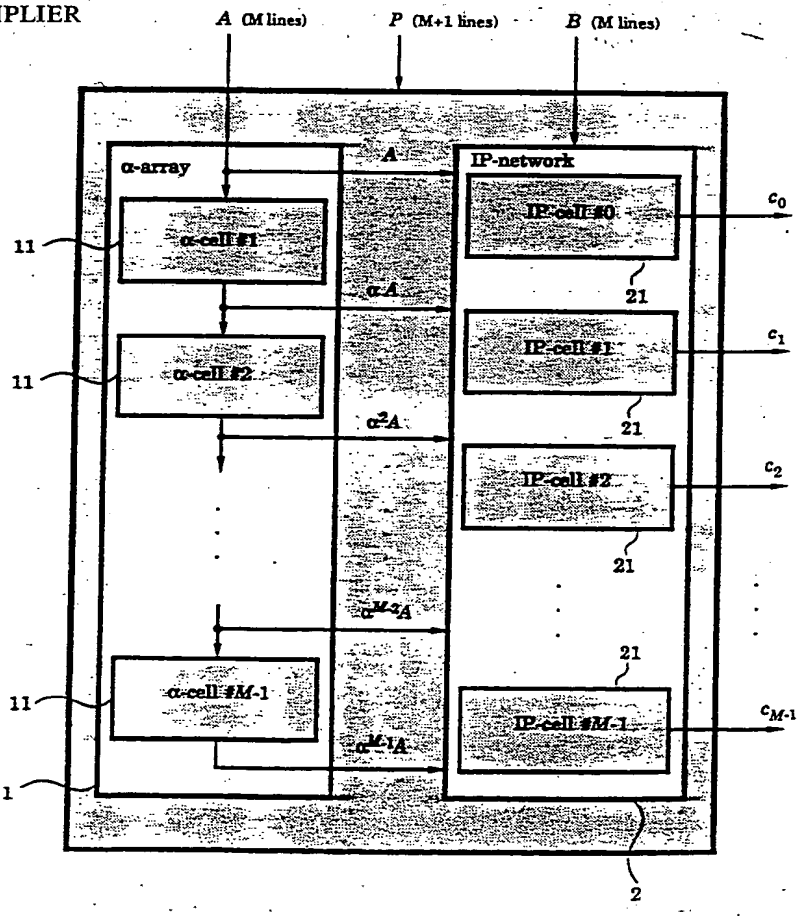
(81) Designated States: AT (European patent), AU, BE (European patent), CA, CH (European patent), DE (European patent), DK (European patent), ES (European patent), FI, FR (European patent), GB (European patent), GR (European patent), IT (European patent), LU (European patent), NL (European patent), NO, SE (European patent), US.

Published
With international search report.

(54) Title: UNIVERSAL GALOIS FIELD MULTIPLIER

(57) Abstract

The present invention provides a novel apparatus for computing products in Galois fields $GF(p^m)$ with emphasis on the case $p = 2$. The elements of the field are represented in polynomial basis and no basis conversion is required. The apparatus consists of two distinct subunits. The first subunit simultaneously produces the first m α-multiples of one of the two elements to be multiplied. The second subunit simultaneously produces the m inner products of the second element and the m vectors consisting of suitable components of the above mentioned α-multiples. Both subunits are capable of operating over any Galois field $GF(p^m)$ where m is an integer in the range [2, M]. Consequently, the apparatus is programmable for operation over any of the above mentioned Galois fields.



FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AT	Austria	**ES**	Spain	**MG**	Madagascar
AU	Australia	**FI**	Finland	**ML**	Mali
BB	Barbados	**FR**	France	**MN**	Mongolia
BE	Belgium	**GA**	Gabon	**MR**	Mauritania
BF	Burkina Faso	**GB**	United Kingdom	**MW**	Malawi
BG	Bulgaria	**GN**	Guinea	**NL**	Netherlands
BJ	Benin	**GR**	Greece	**NO**	Norway
BR	Brazil	**HU**	Hungary	**PL**	Poland
CA	Canada	**IT**	Italy	**RO**	Romania
CF	Central African Republic	**JP**	Japan	**SD**	Sudan
CG	Congo	**KP**	Democratic People's Republic of Korea	**SE**	Sweden
CH	Switzerland			**SN**	Senegal
CI	Côte d'Ivoire	**KR**	Republic of Korea	**SU**	Soviet Union
CM	Cameroon	**LI**	Liechtenstein	**TD**	Chad
CS	Czechoslovakia	**LK**	Sri Lanka	**TG**	Togo
DE	Germany	**LU**	Luxembourg	**US**	United States of America
DK	Denmark	**MC**	Monaco		

Universal galois field multiplier

The invention is concerned with the multiplication of two arbitrary elements belonging to a Galois field, especially an apparatus for performing such multiplication.

Galois fields are finite fields consisting of p^m elements, where p is a prime number and m a positive integer. The field GF(2^m) is of particular importance in practice because its elements can be represented by binary polynomials of degree at most m-1 in a particular primitive element. This primitive element is a root of the irreducible primitive polynomial of degree m that generates the Galois field.

Galois fields are of fundamental importance in the construction, encoding and decoding of several classes of powerful error-control codes (here abbreviated ECC) like Bose-Chaudury-Hocqenhem codes (called BCH codes), Reed-Solomon codes (called RS codes) and Goppa codes. The reader is referred to F.J. MacWilliams, N.J.A. Sloane "The Theory of Error-Correcting Codes", Amsterdam: North-Holland 1977, for details on the theory of ECC and an introduction to the theory of finite fields. The book by R.E. Blahut, "Theory and practice of Error Control Codes", Cambridge, MA:Addison-Wesley, 1984, gives another treatment of the same theories with emphasis on the practical aspects.

The main parameters of an ECC are the block length n, the number of information symbols k (also called the dimension) and the minimum (Hamming) distance d between two any codewords of the code. A code with minimum distance d is capable of correcting t errors and s erasures as long as $2t + s \leq d$-1. ECCs are very useful in practice for improving the reliability of a noisy communication channel. However, different applications require different codes with different parameters n, k, d. These parameters are all directly or indirectly related to the number (=2^m) of elements of the Galois field GF(2^m). For example the maximum block length n of an RS code is $2^m + 1$. This means that, if we are constrained to use one single Galois field we are also limited in our selection of ECC.

Building a dedicated hardware for every code of practical interest is obviously unreasonable. Sometimes dedicated hardware can though be motivated by standardization and/or by extreme speed requirements. In many other situations a flexible, programmable device capable of implementing different codes over different Galois fields would be the most appropriate choice. The most crucial and important single unit in a device capable of providing the aforementioned flexibility, is a fast universal Galois field multiplier (here abbreviated UGM) capable of operating over a

number of different Galois fields. Actually, multiplication is by far the most common operation occurring in the encoding/decoding procedures of, for example, BCH and RS codes. Successive multiplications can also be used to compute the inverse of a field element. Inversion is required in the decoding of, for example, BCH and RS codes.

A prior art UGM has resulted in a cellular-array multiplier which is too slow to be really practical. The poor performance of the prior art UGM is due to a worst signal path of about $6m$ levels of logic when the UGM is operated over GF(2^m). Details on the prior art UGM are found in B.A. Laws, C.K. Rushforth, "A Cellular-Array Multiplier for GF(2^m)", IEEE Trans. Comput., Vol. C-20, pp. 1573-1578, December 1971.

The principal object of the invention is to provide a novel apparatus for computing products of elements belonging to a Galois field GF(p^m) with emphasis on the case $p = 2$. The new apparatus has fewer components and higher speed than previous art apparatus.

It is a feature of this invention to be programmable for operation over any Galois field GF(p^m) with $2 \leq m \leq M$ where M is an arbitrary positive integer greater than one.

The invention, as well as the embodiments thereof, is defined in the appended claims.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a block diagram of apparatus according to a preferred organization.

FIG. 2 is a more detailed block diagram of a sub-unit of apparatus used to compute $\alpha \cdot A$ over different fields of characteristic two.

FIG. 3 is yet a more detailed block diagram of a sub-unit of apparatus used to compute the inner product of two binary vectors.

FIG. 4 is an example of apparatus for the fields GF(2^m), $2 \leq m \leq 4$.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

The discussion of apparatus requires a review of some basic properties of a Galois field. A Galois field GF(p^m) is an algebraic finite field consisting of p^m elements, where p is a prime and m a positive integer. Among the field elements are included the null element, 0, and the unit element, 1. Upon the elements in the field are defined the operations of addition, subtraction, multiplication and division. Addition, subtraction and multiplication are

associative and commutative and multiplication is distributive with respect to addition and subtraction. Further, any of the four aforementioned operations results always in an element of the field.

The present invention is primarily concerned with, but not limited to fields of characteristic two (i.e. $p = 2$) which are denoted by $GF(2^m)$. The smallest of these fields (m=1) consists actually only of a null element 0 and a unit element 1 and it is called the binary field GF(2). Addition and multiplication in GF(2) are performed modulo 2, i.e. 0+0=1+1=0, 0+1=1+0=1, $0 \cdot 0 = 0 \cdot 1 = 1 \cdot 0 = 0$, $1 \cdot 1 = 1$ and -1=1. Addition is thus the same as exclusive-or (XOR) whereas multiplication is the same as logical AND.

In $GF(2^m)$, $m > 1$, each element can be represented by a polynomial of degree m-1 or less with binary coefficients. Each element is a residue modulo an irreducible polynomial of degree m over $GF(2)$, and all arithmetic operations on the coefficients are performed modulo 2. Alternatively, the field $GF(2^m)$ can be seen as a linear vector space over GF(2) of dimension m (in which case it should be denoted $GF(2)^m$).

For each integer m there exists only one finite field with 2^m elements (this is true in general for fields of any characteristic). In general, however, there exist several different *representations* of the elements of a finite field. The particular representation is given by the particular irreducible polynomial chosen to generate the finite field.

Representing an element A as a polynomial $a_0 + a_1x + \ldots + a_{m-2}x^{m-2} + a_{m-1}x^{m-1}$ corresponds to choosing the set of field elements $\{1, \alpha, \ldots, \alpha^{m-2}, \alpha^{m-1}\}$ as a basis of $GF(2^m)$. Every element can thus be expressed as a linear combination of the basis elements. In particular, the elements α^i, $i = 0, 1, \ldots, m-1$ are represented in this basis by the polynomials x^i, i=0, 1, ..., m–1 and the expression $a_0 + a_1x + \ldots + a_{m-2}x^{m-2} + a_{m-1}x^{m-1}$ is equivalent to $a_0 + a_1\alpha + \ldots + a_{m-2}\alpha^{m-2} + a_{m-1}\alpha^{m-1}$. The type of basis discussed above is naturally called the polynomial basis.

In the following we call $P(x)$ the irreducible polynomial generating the field and which has the field element α as a root, i.e. $P(\alpha) = 0$. $A(x)$ is the polynomial associated with the field element A, $B(x)$ the polynomial associated with the field element B and $C(x)$ the polynomial associated with the product of A and B. Then the product is given by the following expression

$$C(x) = A(x) \cdot B(x) \bmod P(x) =$$

$$= [b_0A(x) + b_1xA(x) + \ldots + b_{m-1}x^{m-1}A(x)] \bmod P(x) =$$

$$= [b_0 A(x) \bmod P(x)] + [b_1 xA(x) \bmod P(x)] + \ldots +$$

$$[b_{m-1} x^{m-1} A(x) \bmod P(x)]. \qquad (1)$$

We define now the polynomials $Z_{i,-}(x)$ as follows:

$$Z_{i,-}(x) = \sum_{j=0}^{m-1} z_{i,j} x^j = x^i A(x) \bmod P(x) \quad i = 0, 1, \ldots, m-1 \qquad (2)$$

where $z_{i,j} \in$ GF(2). Then

$$C(x) = b_0 Z_{0,-}(x) + b_1 Z_{1,-}(x) + \ldots + b_{m-1} Z_{m-1,-}(x). \qquad (3)$$

And in matrix notation

$$C = \begin{pmatrix} c_0 \\ c_1 \\ \vdots \\ c_{m-1} \end{pmatrix} = \begin{pmatrix} z_{0,0} & z_{1,0} & \ldots z_{m-1,0} \\ z_{0,1} & z_{1,1} & \ldots z_{m-1,1} \\ \vdots & \vdots & \vdots \\ z_{0,m-1} & z_{1,m-1} & \ldots z_{m-1,m-1} \end{pmatrix} \begin{pmatrix} b_0 \\ b_1 \\ \vdots \\ b_{m-1} \end{pmatrix} = Z \cdot B \qquad (4)$$

where Z is the m by m binary matrix in equation (4). We see that the product C can be obtained by computing the m inner products $Z_{-,j} \cdot B, j = 0,1\ldots, m-1$, where $Z_{-,j}$ denotes the j:th row of Z. First, though, the entries of Z have to be generated and this can be done as follows. We generate the m columns of Z simultaneously by cascading m-1 identical cells where each cell implements the operation $xA(x)$ mod $P(x)$ (the first column $Z_{0,-}$ is the element A itself, see equation (2)). We call such a cell the α-cell and the cascaded structure the α-array.

The polynomial $P(x)$ used to generate the field is of the form $x^m + x^{m-1} p_{m-1} + \ldots + x p_1 + 1$ (the first and last coefficient must necessarily be ones if $P(x)$ is to be irreducible). Then the expression $xA(x)$ mod $P(x)$ can be written as follows:

$$x \cdot A(x) = x^m a_{m-1} + x^{m-1} a_{m-2} + \ldots + x^2 a_1 + x a_0 =$$

$$= a_{m-1}(x^{m-1} p_{m-1} + \ldots + x p_1 + 1) + x^{m-1} a_{m-2} + \ldots + x^2 a_1 + x a_0 =$$

$$= a_{m-1} + \sum_{i=1}^{m-1} x^i (a_{m-1} p_i + a_{i-1}) \qquad \bmod P(x). \qquad (5)$$

In equation (5) we have utilized the fact that $\alpha^m = \alpha^{m-1} p_{m-1} + \ldots + \alpha p_1 + 1$ (or equivalently $x^m = x^{m-1} p_{m-1} + \ldots + x p_1 + 1$). Equation (5) describes the

function of the α-cell for fixed m: for each $p_i \neq 0$, $i = 1, 2, \ldots, m\text{-}1$, one sum $a_{m-1} + a_{i-1}$ has to be computed whereas the coefficient of x^0 is A's most significant coefficient a_{m-1}. We call a_{m-1} the feedback (FB) signal.

Having described the mathematical preliminaries, a preferred embodiment of the novel UGM will now follow.

A. Hardware

Fig. 1 shows the general structure of the novel UGM. The notation is consistent with the previous section. Unit 1 is the α-array that generates the entries of the matrix Z as defined in equation (4). Unit 2 computes the m inner products $c_j = Z_{\cdot,j} \cdot B, j = 0,1\ldots, m\text{-}1$ and is here called the IP network. The IP network consists in turn of m identical cells, where each cell, here called the IP-cell, computes one inner product. The UGM requires the input field elements to have zeros in the unused high-order positions, i.e. $a_i = b_i = 0, i > m\text{-}1$.

Fig. 2 shows a preferred implementation of the α-cell 11 for performing the operation $xA(x)$ mod $P(x)$ (or, equivalently, αA). The α-cell can be *programmed* to operate over any of the fields GF(2^m), $2 \leq m \leq M$ by means of the binary vectors $P = (p_1, p_2, p_3, \ldots, p_{M-1})$ and $S = (s_1, s_2, s_3, \ldots, s_{M-1})$ shown in Fig. 2.

Suppose we want to program the UGM for operation over GF(2^m) where m is a particular value in the usable range. Then the components of the vector S are set as follows:

$$s_i = \begin{cases} 1 & i = m\text{-}1 \\ 0 & i \neq m\text{-}1 \end{cases} \qquad (6)$$

The vector S determines the feedback signal FB of Fig. 2. The first m-1 components of the vector P are the m-1 middle coefficients of the irreducible polynomial $P(x)$ chosen to generate the field. The remaining coefficients p_m through p_{M-1} are, for example, set to zero.

We see in Fig. 2 that the α-cell has a regular bit-slice structure consisting of m-1 identical subcells (unit 111 in Fig. 2). In each subcell there is one binary adder (XOR), one switch SW and one multiplexer MX. The switch SW in subcell #i is controlled by the signal s_i in the following way: SW is closed if $s_i = 1$, SW is open if $s_i = 0$. The multiplexer MX is controlled by the signal p_i in the following way: if $p_i = 1$ then MX passes the signal coming from the binary adder ($= a_{m-1} + a_{i-1}$), if $p_i = 0$ then MX passes the other input ($= a_{i-1}$).

Fig. 3 shows a preferred implementation of the IP-cell 21 based on two-input gates. M AND gates and M-1 XOR gates are required. The multiplexer MX appended to the output of the IP-cell is required to zero the product coefficients c_i for $i > m$ -1 since these are not used. In this case the signal v_i is the i:th component of a vector $V = (v_0, v_1, ..., v_{M-1})$ that could be set as follows

$$v_i = \begin{cases} 0 & i \le m\text{-}1 \\ 1 & i > m\text{-}1 \end{cases} \qquad (7)$$

The multiplexer MX would then zero the output if $v_i = 1$. If $v_i = 0$ the output of the XOR-tree is selected.

Fig. 4 shows the complete UGM for the case $M = 4$ together with a table of values for the vectors S and V for $2 \le m \le 4$. Notice that $m \ge 2$ implies that the first two components s_0 and s_1 of S are always zero and need not be generated (the multiplexer could be skipped in those IP-cells). The field generator $P(x)$ is not indicated but can be chosen as follows: $P(x) = x^4 + x + 1$ for $m = 4$, $P(x) = x^3 + x + 1$ for $m = 3$ and $P(x) = x^2 + x + 1$ for $m = 2$.

The extension to a new value of M is straightforward.

Operating the UGM for $m < M$ means that only a part of α-array is used. This fact can be easily illustrated by help of equation (4). First we define the vectors C_L, C_U, B_L and B_U as follows

$$C_L = (c_0, ..., c_{m-1})^T \qquad C_U = (c_m, ..., c_{M-1})^T$$

$$B_L = (b_0, ..., b_{m-1})^T \qquad B_U = (b_m, ..., b_{M-1})^T$$

where the superscript T indicates transposition. Then we have

$$\begin{pmatrix} C_L \\ C_U \end{pmatrix} = \left(\begin{array}{ccc|ccc} z_{0,0} & \cdots & z_{m-1,0} & z_{m,0} & \cdots & z_{M-1,0} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ z_{0,m-1} & \cdots & z_{m-1,m-1} & z_{m,m-1} & \cdots & z_{M-1,m-1} \\ \hline z_{0,m} & \cdots & z_{m-1,m} & z_{m,m} & \cdots & z_{M-1,m} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ z_{0,M-1} & \cdots & z_{m-1,M-1} & z_{m,M-1} & \cdots & z_{M-1,M-1} \end{array}\right) \begin{pmatrix} b_0 \\ \vdots \\ b_{m-1} \\ b_m \\ \vdots \\ b_{M-1} \end{pmatrix} = \left(\begin{array}{c|c} Z_1 & \\ \hline Z_2 & Z_3 \end{array}\right) \begin{pmatrix} B_L \\ B_U \end{pmatrix} \qquad (8)$$

where Z_1, Z_2 and Z_3 are submatrices of Z defined according to the subdivision of Z indicated in equation (8). The product of interest for us is $Z_1 \cdot B_L$ and we want it to appear on the lines of C_L. To have this product

correctly computed we must ensure that the product $Z_3 \cdot B_U$ is always zero. But this is the case since B_U is required to be zero. What remains to take care of is the product $Z_2 \cdot B_L$ since this is normally non-zero and it would appear on the lines of C_U (the unused lines that we wish to be zero). The zeroing of these lines is done through the multiplexer MX and the control signal v_i mentioned above and shown in Fig. 3.

B. Complexity

The α-array consists of m-1 α-cells where each cell contains m-1 XOR gates, m-1 switches and m-1 multiplexers. Since switches and multiplexers are much simpler than XOR gates we approximate the complexity of a switch-multiplexer pair by that of one XOR gate. Then the complexity of the α-array can be estimated to $2(m-1)^2$ gates. The IP-network consists of m IP-cells where each cell contains $2m$-1 gates. Totally $2m^2$-m gates for the IP-network. Finally we need $3m$ register to store the vectors P, S and V needed to program the UGM (these registers are loaded from an external unit). The complexity N_{UGM} for the whole UGM can therefore be estimated by

$$N_{UGM} \approx 2(m-1)^2 + 2m^2-m + 3m.$$

Compared to a prior art UGM with complexity ~ $7m^2+3m$ the present UGM requires about 50% less components.

C. Performance

The performance of the UGM is directly related to the worst signal path (WSP) between any input and any output of the UGM. We will give an upper bound on the length L_{WSP} (in gates) of the WSP. In doing this we approximate the delay of a switch-multiplexer pair by that of one XOR gate.

The WSP through the UGM must go through m-1 α-cells and one IP-cell.

The length of the WSP through the IP-cell is fixed and it is easily found to be $1+\lceil \log_2 M \rceil$ gates.

The WSP through the α-array depends on the choice of $P(x)$. It consists however of three parts: switches, XOR gates and multiplexers. The number of XOR gates along the WSP can be much less than m-1 by smart choice of $P(x)$. The following is a table over the number of XOR gates along the WSP through the α-array for some good $P(x)$ and $m \leq 16$:

m	$P(x)$	# of XOR gates
2	2,1,0	1
3	3,1,0	1
4	4,1,0	1
5	5,2,0	2
6	6,1,0	1
7	7,1,0	1
8	8,5,3,2,0	4
9	9,4,0	2
10	10,3,0	2
11	11,2,0	2
12	12,8,5,1,0	4
13	13,7,6,1,0	4
14	14,9,7,2,0	4
15	15,1,0	1
16	16,11,6,5,0	5

In the table we indicate only the powers of x in $P(x)$ whose coefficients are non-zero . We see that the number of XOR gates is at least one and at most $\frac{m}{2}$ for $m \leq 8$. For $m > 8$ a better upper bound seems to be $\frac{m}{3}$. We use $\frac{m}{2}$ as an upper bound for all m.

The number of switches and multiplexers along the WSP is not easily determined exactly. We assume worst case and say therefore that the WSP goes through m-1 switches and m-1 multiplexers. According to the approximation above this corresponds to about m-1 XOR gates.

The total length L_{WSP} of the WSP can now be upper bounded by

$$L_{WSP} \leq (m\text{-}1) + m/2 + 1 + \lceil \log_2 M \rceil = 1.5m + \lceil \log_2 M \rceil \text{ [Gates]}$$

which is considerably better than the ~ $6m$ gates of a prior art UGM.

D. Comments

One skilled in the art will immediately recognize that several changes could be made in the above design without departing form the basic structure. For example, instead of storing the three vectors P, S and V in registers one could design some simple logic that generates both S and V from P (in this case also the highest coefficient p_m of $P(x)$ must be entered into the UGM). The programming of the UGM would thus be simplified to one single operation instead of three. The UGM is also easily modified to perform the operation $A \cdot B + D$ by adding one input and one XOR gate to each IP-cell. Further, the design of the sub-cell 111 can alternatively be done by using an AND gate instead of the multiplexer MX. The AND gate

computes the product $a_{m-1}p_i$. This product enters then the XOR gate (instead of the feedback signal a_{m-1}) to produce the sum $a_{m-1}p_i + a_{i-1}$.

The same general structure of Fig. 1 can be utilized for UGMs operating over fields of characteristic other than two. Only the details get slightly more complicated since all coefficient operations must be performed modulo the prime p, $p > 2$, that is the XOR gate becomes a mod p-adder and the AND gate a mod p-multiplier. Further, for prime $p > 2$ we have $-1 \neq 1$ mod p which means that signs must be considered. For example, suppose $P(x)$ is a monic (i.e. with the highest coefficient $p_M = 1$) irreducible polynomial of degree M over GF(p) that has α as a root, i.e. $P(\alpha) = 0$. Then

$$\alpha^M = -\alpha^{M-1}p_{M-1} - \ldots - \alpha p_1 - p_0 = \alpha^{M-1}p'_{M-1} + \ldots + \alpha p'_1 + p'_0 \qquad (9)$$

where p'_i is the additive inverse of p_i in GF(p). Now equation (5) becomes

$$x \cdot A(x) = x^M a_{M-1} + x^{M-1}a_{M-2} + \ldots + x^2 a_1 + x a_0 =$$

$$= a_{M-1}(-x^{M-1}p_{M-1} - \ldots - xp_1 - p_0) + x^{M-1}a_{M-2} + \ldots + x^2 a_1 + x a_0 =$$

$$= p'_0 a_{M-1} + \sum_{i=1}^{M-1} x^i (p'_i a_{M-1} + a_{i-1}) \mod P(x). \qquad (10)$$

The design of the α-cell follows directly from equation (10). The α-cell consists of M-1 identical sub-cells where each sub-cell performs the operation $p'_i a_{M-1} + a_{i-1}$ plus one cell for computing $p'_0 a_{M-1}$, where juxtaposition means modulo p-multiplication and "+" modulo p-addition. Since P is known in advance the additive inverses p'_i, $i = 0, 1, 2, \ldots, m-1$ can be precomputed and input to the multiplier instead of the original coefficients p_i. The α-cell is made programmable for operation over different fields GF(p^m), $2 \leq m \leq M$ just the same way as for $p = 2$ by means of switches and the control vector S. The new α-array is obtained simply by cascading M-1 α-cells just as before. The α-array is connected to the IP network as before to compute the necessary inner products. The IP cell is modified to compute the inner product of two p-ary vectors of length M. The control vector V is used as for $p = 2$. The vectors S and V can either be stored in registers which are loaded from outside or they can be derived from the coefficients of P (in fact only the position of the highest coefficient p_m is relevant to this purpose) by some simple logic. We notice finally that the binary representation of each coefficient will require $\lceil \log_2 p \rceil$ bits. For example the three elements of GF(3) require two bits.

Accordingly, it is intended that all matter contained in the above descriptions and the following drawings shall be interpreted as illustrative and not in a limiting sense.

CLAIMS

1. A multiplier for performing multiplication of two elements in the finite field GF(p^m) with p^m elements, and obtaining a product vector of m p-ary components, where m is an integer equal to or greater than 2 or equal to or less than M, where M is an integer equal to or greater than 2, each of said p^m elements of GF(p^m) represented by a vector of m p-ary coefficients according to a polynomial basis representation, c h a r a c t e r i z e d b y

a) first logic means (1) including a cascade of at least one α-cell (11) for developing for the first of said two elements the first m α-multiples, each α-multiple being the product of α^i and said element for $i = 0, 1, 2, \ldots, m\text{-}1$, where α is an element of the field GF(p^m) satisfying the equation $P(x) = 0$ for $x = \alpha$, where $P(x)$ is a polynomial of degree m which is irreducible over the field GF(p); and

b) second logic means (2) including at least two IP cells (21), where each IP cell will simultaneously develop the inner product of the second element and every p-ary vector whose components are the j:th components of all said α-multiples for $j = 0, 1, 2, \ldots, m\text{-}1$, each of said m inner products being one component of said product vector.

2. The multiplier recited in claim 1 w h e r e i n:

a) said first logic means (1) comprise means for changing of said irreducible polynomial, whereby said first logic means are programmable for operation over any of said finite fields GF(p^m), $2 \leq m \leq M$, including all possible representations of said finite fields; and

b) means for selectively connecting the output of said second logic means (2) to a logical zero.

3. The multiplier recited in claim 1 or 2 w h e r e i n each of the p^m elements of GF(p^m) is represented by a vector of m p-ary components according to a polynomial basis representation of the form $A = a_0 + a_1\alpha + \ldots + a_{m-2}\alpha^{m-2} + a_{m-1}\alpha^{m-1}$, where A is an element of GF(p^m), $a_0, a_1, \ldots, a_{m-2}, a_{m-1}$ are the p-ary components of A, and α is an element of GF(p^m)

satisfying the equation $P(x) = 0$ for $x = \alpha$, where $P(x)$ is a polynomial of degree m which is irreducible over the field GF(p).

4. The multiplier recited in claim 2 or 3 wherein:

a) the unused inputs of said first logic means (1) are set to logical zero; and

b) the unused inputs and outputs of said second logic means (2) are set to logical zero.

5. The multiplier recited in claim 1, 2, 3 or 4 wherein $p = 2$.

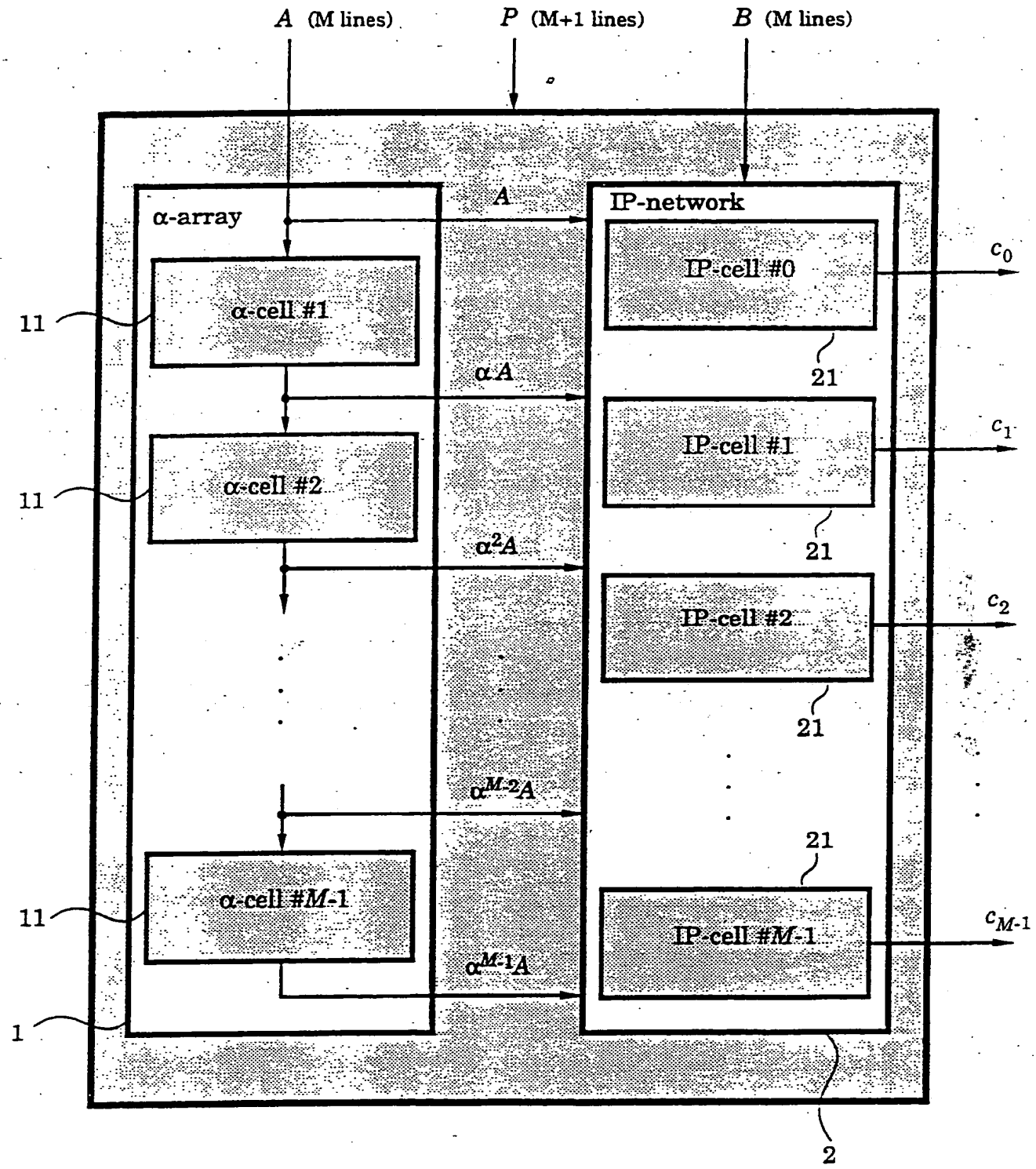


FIG. 1

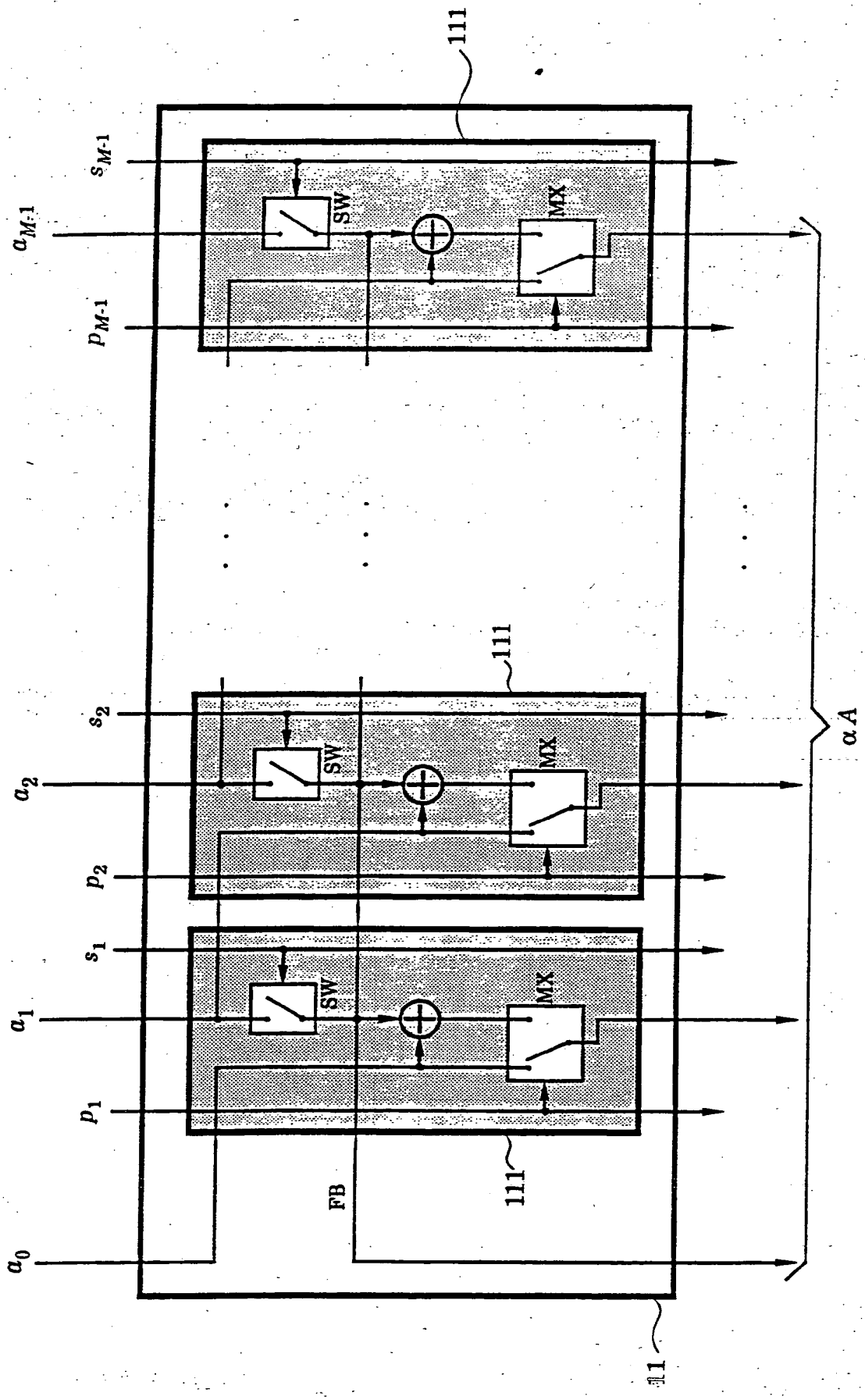


FIG. 2

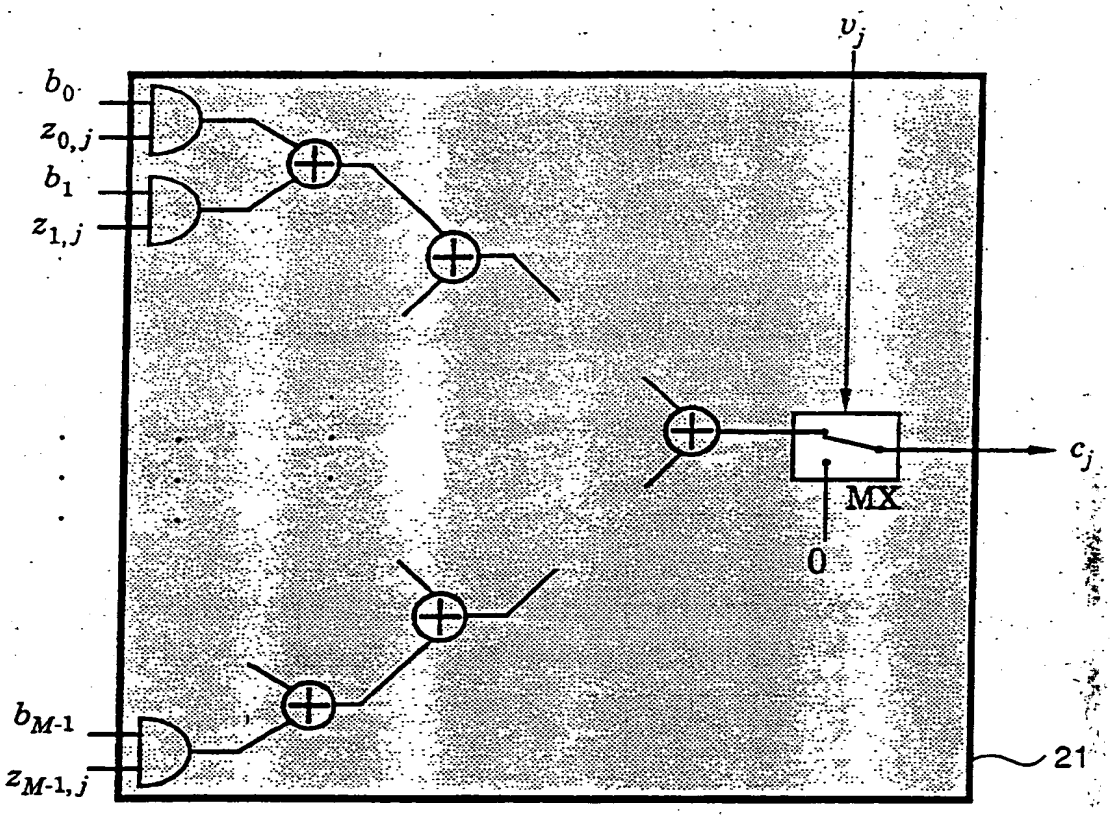
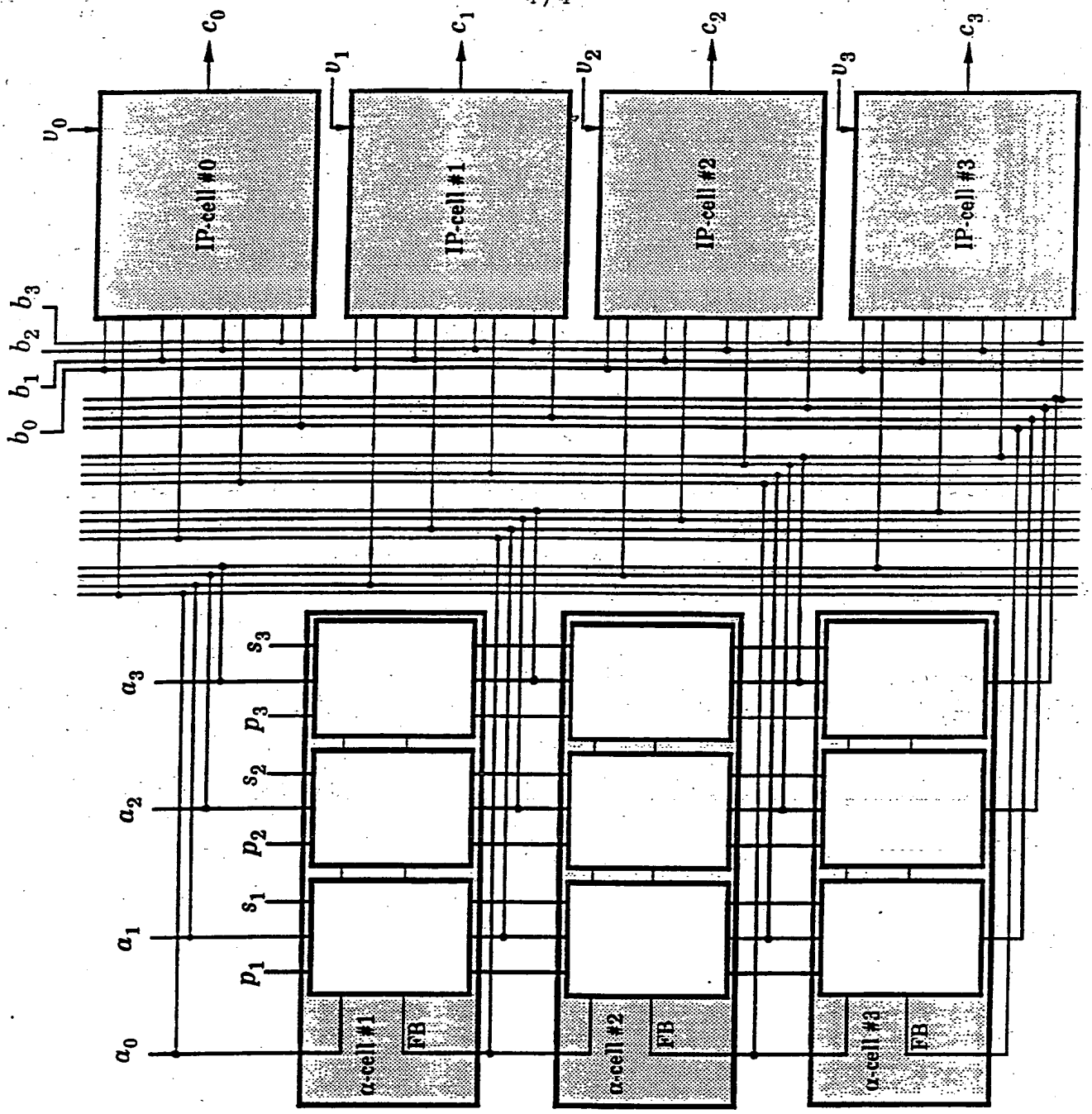


FIG. 3



m	s_1	s_2	s_3	v_0	v_1	v_2	v_3
2	1	0	0	0	0	1	1
	0	1	0	0	0	0	1
3	0	0	1	0	0	0	0
4							

FIG. 4

INTERNATIONAL SEARCH REPORT

International Application No PCT/SE 91/00384

I. CLASSIFICATION OF SUBJECT MATTER (if several classification symbols apply, indicate all)[6]

According to International Patent Classification (IPC) or to both National Classification and IPC

IPC5: G 06 F 7/52

II. FIELDS SEARCHED

Minimum Documentation Searched[7]

Classification System	Classification Symbols
IPC5	G 06 F

Documentation Searched other than Minimum Documentation to the Extent that such Documents are Included in Fields Searched[8]

SE,DK,FI,NO classes as above

III. DOCUMENTS CONSIDERED TO BE RELEVANT[9]

Category *	Citation of Document,[11] with indication, where appropriate, of the relevant passages[12]	Relevant to Claim No.[13]
A	US, A, 3805037 (J. T. ELLISON) 16 April 1974, see the whole document --	1-5
A	US, A, 4251875 (J.M. MARVER ET AL) 17 February 1981, see the whole document --	1-5
A	US, A, 4697248 (N. SHIROTA) 29 September 1987, see the whole document --	1-5

---

* Special categories of cited documents: [10]

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance, the claimed invention cannot be considered novel or cannot be considered to involve an inventive step

"Y" document of particular relevance, the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

IV. CERTIFICATION

Date of the Actual Completion of the International Search	Date of Mailing of this International Search Report
11th September 1991	1991 -09- 17
International Searching Authority	**Signature of Authorized Officer**
SWEDISH PATENT OFFICE	ANDERS HOLMBERG

Form PCT/ISA/210 (second sheet) (January 1985)

ANNEX TO THE INTERNATIONAL SEARCH REPORT
ON INTERNATIONAL PATENT APPLICATION NO.PCT/SE 91/00384

This annex lists the patent family members relating to the patent documents cited in the above-mentioned international search report.
The members are as contained in the Swedish Patent Office EDP file on 91-07-31
The Swedish Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US-A- 3805037	74-04-16	DE-A- 2306323	73-08-30
US-A- 4251875	81-02-17	CA-A- 1120595	82-03-23
US-A- 4697248	87-09-29	EP-A-B- 0152702	85-08-28
		JP-A- 60144834	85-07-31